#include &lt;iostream&gt;

using namespace std;

int board[8][8] = { 0 };//tabela me permasa 8 me 8

int tries = 0;//variabel qe numeron perpjekjet

int failed = 0;//variabel qe numeron deshtimet

//-----PrototypeFunctions----------------

void printBoard(int row, int col);

bool moveKnight(int row, int col, int count);

int main() {

//-------LocalVariables------------------

int currentRow = 8; //size of board

int currentCol = 8; //Size of board

int knightX = 0; //koordinatat e pozicionit me te larte majtas.

int knightY = 0; //koordinatat e pozicionit me te larte majtas.

int move = 1; // shenon fillimin e levizjeve me 1 per levizjen fillestare.

//--------BeginTryingToMove--------------

cout &lt;&lt; &quot;Tabela e Kaloresit&quot; &lt;&lt; endl;

printBoard(currentRow, currentCol);//printn tabelen e 0

moveKnight(knightX, knightY, move);//thirrja e funksionit.

//shfaq numrin e perpjekjeve dhe deshtimeve:

cout &lt;&lt; &quot;Kaloresi kryen nje udhetim te plote :&quot; &lt;&lt; endl

&lt;&lt; &quot;udhetimit te tij iu deshen &#39; &quot; &lt;&lt; tries &lt;&lt; &quot; &#39; here per te gjetur&quot;

&lt;&lt; endl &lt;&lt; &quot;rrugen e vetme te sakte dhe &#39; &quot; &lt;&lt; failed

&lt;&lt; &quot; &#39; here deshtoi&quot; &lt;&lt; endl;

printBoard(currentRow, currentCol);//printon tabelen me 0.

system(&quot;pause&quot;);

return 0;

}

//-----------Functions------------------

//PrintBoard :

// \*\* ky funksion perdor permasat e tabeles dhe tabelen globale

// te inicializuar per ta paraqitur ate.\*\*

void printBoard(int horizontal, int vertical) {

//printon vektorin dydimensional duke perdorur rrjeshtat dhe shtyllat e meperparshme:

for (int k = 0; k &lt; vertical; k++) {

for (int j = 0; j &lt; horizontal; j++) {

cout &lt;&lt; board[k][j] &lt;&lt; &#39; &#39;;

}

cout &lt;&lt; endl;

}

}

//Leviz kaloresin :

// \*\* ky funksion Boolean kontrollon per drejtimet nga ku

// duhet te leviz kaloresi.\*\*

bool moveKnight(int row, int col, int count) {

//Lejon te leviz nese kaloresi ndodhet ne tabele dhe ne nje vend te lire.

if (row &gt;= 0 &amp;&amp; row &lt; 8 &amp;&amp; col &gt;= 0 &amp;&amp; col &lt; 8 &amp;&amp; board[row][col] == 0) {

board[row][col] = count;//I jep cdo pozicioni nga nje numer.

if (count == 64) {

return true;

}

}

else {

failed = failed + 1; // rrit numrin e perpjekjeve te deshtuara me 1.

return false;//kthehet pas dhe kontrollon serisht

}

//---------Levizja 1---------------------------

if (moveKnight(row - 2, col + 1, count + 1) == 1) {//nqs levizja e pare eshte e mundur

return true; //dhe brenda tabeles.

}

//----------Levizja 2---------------------------

if (moveKnight(row - 1, col + 2, count + 1) == 1) {//nqs levizja e dyte eshte e mundur

return true; //dhe brenda tabeles.

}

//----------Levizja 3---------------------------

if (moveKnight(row + 1, col + 2, count + 1) == 1) { //nqs levizja e trete eshte e mundur

return true; //dhe brenda tabeles.

}

//-----------Levizja 4--------------------------

if (moveKnight(row + 2, col + 1, count + 1) == 1) { //nqs levizja e katert eshte e mundur

return true; //dhe brenda tabeles.

}

//------Levizja 5---------------------------

if (moveKnight(row + 2, col - 1, count + 1) == 1) { //nqs levizja e peste eshte e mundur

return true; //dhe brenda tabeles.

}

//------Levizja 6---------------------------

if (moveKnight(row + 1, col - 2, count + 1) == 1) { //nqs levizja e gjashte eshte e mundur

return true; //dhe brenda tabeles.

}

//-------Levizja 7---------------------------

if (moveKnight(row - 1, col - 2, count + 1) == 1) { //nqs levizja e shtate eshte e mundur

return true; //dhe brenda tabeles.

}

//--------Levizja 8---------------------------

if (moveKnight(row - 2, col - 1, count + 1) == 1) { //nqs levizja e tete eshte e mundur

return true; //dhe brenda tabeles.

}

else {

board[row][col] = 0;//cshenjon pozicionin.

tries = tries + 1;//rrit numrin e perpjekjeve me 1.

return false;//kthen hapin e fundit dhe vlerat.

}

}